



Frequently Asked Questions About Caché Basic

Version 5.2
01 September 2006

Frequently Asked Questions About Caché Basic

Caché Version 5.2 01 September 2006

Copyright © 2006 InterSystems Corporation.

All rights reserved.

This book was assembled and formatted in Adobe Page Description Format (PDF) using tools and information from the following sources: Sun Microsystems, RenderX, Inc., Adobe Systems, and the World Wide Web Consortium at www.w3c.org. The primary document development tools were special-purpose XML-processing applications built by InterSystems using Caché and Java.



The Caché product and its logos are registered trademarks of InterSystems Corporation.



The Ensemble product and its logos are registered trademarks of InterSystems Corporation.



The InterSystems name and logo are trademarks of InterSystems Corporation.

This document contains trade secret and confidential information which is the property of InterSystems Corporation, One Memorial Drive, Cambridge, MA 02142, or its affiliates, and is furnished for the sole purpose of the operation and maintenance of the products of InterSystems Corporation. No part of this publication is to be used for any other purpose, and this publication is not to be reproduced, copied, disclosed, transmitted, stored in a retrieval system or translated into any human or computer language, in any form, by any means, in whole or in part, without the express prior written consent of InterSystems Corporation.

The copying, use and disposition of this document and the software programs described herein is prohibited except to the limited extent set forth in the standard software license agreement(s) of InterSystems Corporation covering such programs and related documentation. InterSystems Corporation makes no representations and warranties concerning such software programs other than those set forth in such standard software license agreement(s). In addition, the liability of InterSystems Corporation for any losses or damages relating to or arising out of the use of such software programs is limited in the manner set forth in such standard software license agreement(s).

THE FOREGOING IS A GENERAL SUMMARY OF THE RESTRICTIONS AND LIMITATIONS IMPOSED BY INTERSYSTEMS CORPORATION ON THE USE OF, AND LIABILITY ARISING FROM, ITS COMPUTER SOFTWARE. FOR COMPLETE INFORMATION REFERENCE SHOULD BE MADE TO THE STANDARD SOFTWARE LICENSE AGREEMENT(S) OF INTERSYSTEMS CORPORATION, COPIES OF WHICH WILL BE MADE AVAILABLE UPON REQUEST.

InterSystems Corporation disclaims responsibility for errors which may appear in this document, and it reserves the right, in its sole discretion and without notice, to make substitutions and modifications in the products and practices described in this document.

Caché, InterSystems Caché, Caché SQL, Caché ObjectScript, Caché Object, Ensemble, InterSystems Ensemble, Ensemble Object, and Ensemble Production are trademarks of InterSystems Corporation. All other brand or product names used herein are trademarks or registered trademarks of their respective companies or organizations.

For Support questions about any InterSystems products, contact:

InterSystems Worldwide Customer Support

Tel: +1 617 621-0700

Fax: +1 617 374-9391

Email: support@InterSystems.com

Table of Contents

Frequently Asked Questions About Caché Basic	1
General	1
Programming	2
Support	5
List of Tables	
Caché Basic Syntax Elements	2
Caché Basic and Objects Quick Reference	3

Frequently Asked Questions About Caché Basic

General

Does Caché Basic work only on Windows platforms?

No, Caché Basic is completely platform-independent and works on any platform on which Caché runs. Caché Basic is built into the Caché kernel in the same way as Caché ObjectScript and is not dependent on any external script engine, such as the Microsoft Scripting Host.

Is Caché Basic slower than Caché ObjectScript?

No, Caché Basic source is compiled into the same object code as Caché ObjectScript, so the performance of Caché Basic is the same (with minor variations) as that of Caché ObjectScript.

On what breed of Basic is Caché Basic based?

The syntax of Caché Basic is based on Microsoft VBScript. However, since Caché Basic is a server-side scripting language, tightly integrated with the database, there are a number of differences between these two implementations. For instance, there are no **MsgBox** or **InputBox** functions in Caché Basic, but there are a number of enhancements; for example, the ability to work directly with Caché-specific datatypes, such as globals, lists etc.

Does the introduction of Caché Basic mean that InterSystems is planning to drop support of Caché ObjectScript?

No, the main goal of Caché Basic is to ease the Caché learning curve for developers already familiar with Basic implementations such as Microsoft Visual Basic, not to replace Caché ObjectScript. Both Caché ObjectScript and Caché Basic will be supported and coexist indefinitely.

Are there any other goals beyond “easing the learning curve?”

Yes, with the introduction of Caché Basic, companies using Caché will gain a number of benefits, including:

- Ease in hiring and educating new developers.
- Ease in selling applications developed with Caché. End users do not need to study a new language to use and enhance your applications.
- Re-hosting business logic of VB and ASP applications in Caché. You can migrate the business logic of existing applications from the VB client side or middle tier into Caché, achieving greater performance and scalability. The combination of CSP and Caché Basic is a good candidate for migrating Active Server Pages (ASP) applications.
- Access to interesting resources. By browsing source code archives for VB, Caché developers can find a multitude of valuable and entertaining code samples (i.e. games.) Good starting points are: <http://www.planet-source-code.com> and <http://www.vbcode.com>.

Programming

Is there a “Caché Basic shell” ?

No, but nothing prevents you from implementing your own. Contact an InterSystems representative for an example of a Caché Basic shell.

How do I work with globals in Caché Basic?

You can directly reference globals using the following syntax:

Caché Basic Syntax Elements

Syntax element	Description
<code>^Global(index)="Value" Println ^Global(index)</code>	direct access to globals
<code>Exists(^a(1))</code>	if value defined
<code>Traverse(^a(""))</code>	move to the next or previous subscript
<code>EraseValue, EraseArray, Merge, Lock, Unlock, Increment()</code>	miscellaneous functions

How do I work with Caché-specific structures such as [List](#) and [Piece](#)?

There are a number of enhancements in Caché Basic for [List](#) and [Piece](#) support. For example:

```
l = ListBuild("blue","red")
Println List(l,1)

p = "blue^red"
noOfItems = Len(p,"^")
Println Piece(p,"^",1)
```

Is there any analogue to \$Order in Caché Basic?

The [Traverse\(\)](#) function provides the same functionality as **\$Order** in Caché ObjectScript.

```
i = Traverse(^MyData(""))
While (i <> "")
    Println ^MyData(i)
    i = Traverse(^MyData(i))
Wend
```

How do I work with objects in Caché Basic?

Caché Basic and Objects Quick Reference

Caché Basic syntax	Objects reference
"Basic.Human".ClassMethod()	call class method
obj.Report()	instance method/property
Me.Name="Anton"	current object property/method
obj=New Basic.Human()	create new object
obj=OpenId Basic.Human(1)	open object instance

What is the analogue to ##this.Method() or ..Method() in Caché Basic?

```
Me.Method()
```

How do I work with SQL in Caché Basic?

Use a dynamic query object:

```
result = New %ResultSet()  
result.Prepare("SELECT Name, Age FROM Basic.Human WHERE Age<?")  
result.Execute(Arg1)  
  
While (result.Next())  
    PrintLn result.Data("Name") & ", " & result.Data("Age")  
Wend
```

How do I work with files in Caché Basic?

Use %File object:

```
file = New %File("c:\test.txt")  
file.Open("WSN")  
file.WriteLine("This is a test")  
file.Close()
```

How do I trap errors in Caché Basic?

Use the [On Error Goto](#) statement:

```
Function ErrorTest(Arg1)  
    On Error Goto errorHandler  
    return 1/Arg1  
  
errorHandler:  
    PrintLn "Error ", Err.Number, " ", Err.Description, " ", Err.Source  
    Err.Clear  
    return 0  
  
End Function
```

Why don't I get an <UNDEFINED> error in Caché Basic?

In Basic, each variable is the empty string by default, so instead of an <UNDEFINED> error you get "" when referring to an undefined variable or function.

Use the [Option Explicit](#) statement to avoid inadvertently referencing undefined variables and functions.

Can I call Caché ObjectScript programs from Caché Basic and vice versa?

Yes, you can call both methods and functions/procedures, regardless of the language they are written in.

In Caché ObjectScript:

```
do Procedure^BasicRoutine()  
do ##class(MyClass).BasicClassMethod()
```

In Caché Basic:

```
Procedure@ObjectScriptRoutine()  
"MyClass".ObjectScriptMethod()
```

Can I create SQL triggers in Caché Basic?

No, not in Caché 5, but this feature is planned for a future release.

How do I call \$ZU functions from Caché Basic?

You cannot call them directly. Create a wrapper class with class methods written in Caché ObjectScript and then call these methods from Caché Basic.

Why can't I see local variables outside of the scope of my procedure?

This is correct behavior; the Basic language defines this functionality. The scope of all variables is limited to the procedure or function where it was defined. Exceptions are variable names that start with the % symbol, such as *%myvar*.

How can I convert date and time values to and from \$H format?

Available conversion functions are: **DateConvert()**, **TimeConvert()**, and **DateTimeConvert()**. For example:

```
myDate = "07/15/2002"  
println DateConvert(myDate, vbToInternal)    ' returns $H format  
myHDate = 59000  
println DateConvert(myHDate, vbToExternal)
```

Can I use indirection in Caché Basic?

No.

Support

I suspect there is a bug in Caché Basic. What should I do?

Send a description of the bug to support@intersystems.com.

I need a particular feature in Caché Basic. Can InterSystems implement this?

Caché Basic is ready to use, but still open for enhancements. Send your ideas to andreas@intersystems.com. If your idea is reasonable, InterSystems may decide to implement it in a future version.

I have comments and suggestions for this document. What should I do?

Send it to support@intersystems.com.